Avocado price prediction
Project Overview

Business Objective
Hass avocados, a Mexico based company produces a variety of Avocados which are sold in the US. They have been having good success for the past several years and want to expand. For this, they want to build and assess a plausible model to predict the average price of Hass avocado to consider the expansion of different types of Avocado farms that are available for growing in other regions.

Aim
Forecast the prices of Avocado in the US

Data
The data comes directly from retailers' cash registers based on the actual retail sales of Hass avocados.

- Data represents weekly retail scan data for National retail volume (units) and price from Apr 2015 to Mar 2018.
- The Average Price (of avocados) in the table reflects a per unit (per avocado) cost, even when multiple units (avocados) are sold in bags.
- The Product Lookup codes (PLU's) in the table are only for Hass avocados. Other varieties of avocados (e.g. greenskins) are not included in this table.

Some relevant columns in the dataset:

1. Date - date of the observation
2. AveragePrice - average price of a single avocado
3. Type - conventional / organic
4. Region - region of the observation
5. Total Volume - Total number of avocados sold
6. 4046 - Total number of avocados with PLU 4046 sold
7. 4225 - Total number of avocados with PLU 4225 sold
8. 4770 - Total number of avocados with PLU 4770 sold
9. Total Bags – Total bags sold
10. Small/Large/XLarge Bags – Total bags sold by size

There are two types of avocados in the dataset as well as several different regions represented. This allows you to do all sorts of analysis for different areas of the United States, specific cities, or just the overall United States on either type of avocado. Our analysis will be focused on the complete dataset.

Dataset : https://www.kaggle.com/neuromusic/avocado-prices#avocado.csv

Tech Stack

➔ Language used : Python
➔ Libraries used : statmodels, pmdarima, fbprophet, scikit-learn

Approach

1. Data Preprocessing
    a. Check for missing values
    b. Label Encoding
    c. One hot encoding
2. Exploratory Data Analysis
    a. Identifying any overarching trend in data over time
    b. Identifying any repetitive, seasonal patterns in the data
3. Feature Engineering
    a. Creating new columns
4. Building Forecast models
    a. Linear Regression
    b. Random Forest Regressor
    c. XGB Regressor
    d. Facebook Prophet
    e. ARIMA
    f. SARIMAX
5. Evaluating Forecast models
    a. R-squared
    b. MAPE
    c. MAE
    d. Plots comprising the actual values, forecast and confidence intervals.

Modular Code Overview

The ipython notebook is modularized into different functions so that the user can use those functions instantly whenever needed. The modularized code folder is structured in the following way.

```
input
  |__avocado.csv
src
  |__engine.py
  |__ML_pipeline
           |__arima.py
           |__dataset.py
           |__encoding.py
           |__get_best_arima_params.py
           |__prophet.py
           |__regression_models.py
           |__train_test_split.py
lib
  |__Avocado.html
  |__Avocado.ipynb
output
  |__lin_reg.pkl
  |__rf_reg.pkl
  |__xgb_reg.pkl
```

Once you unzip the modular_code.zip file you can find the following folders within it.

1. input
2. src
3. output
4. lib

1. The input folder contains all the data that we have for analysis.
2. The src folder is the heart of the project. This folder contains all the modularized code for all the above steps in a modularized manner. It further contains the following.
    a. ML_pipeline
    b. engine.py

The ML_pipeline is a folder that contains all the functions put into different python files which are appropriately named. These python functions are then called inside the engine.py file. The deployment folder contains all the files related to deploying the model

3. The output folder contains all the models that we trained for this data saved as reusable files. These models can be easily loaded and used for future use and the user need not have to train all the models from the beginning.
4. The lib folder is a reference folder. It contains the original ipython notebook that we saw in the videos.

Project Takeaways

- Understanding the problem statement
- Importing the necessary libraries and understanding its use
- Importing the dataset
- Performing basic EDA and checking for the null values
- Filling the null values using appropriate methods
- Finding median, average and merging the data
- Feature engineering with the date
- Plotting time-series graphs for visualization
- Drawing a heatmap with the numeric values using Seaborn
- Finding lag of a time series
- Using groupby function for combined analysis of variables
- Differentiating a time series
- Performing train_test_split to divide the dataset into train and test
- Using mean_absolute_percentage_score and mean_absolute_error as evaluation metrics
- Using Adaboost Regressor for making predictions
- Applying the ARIMA time series model for training and making predictions
- Applying Facebook Prophet model for making predictions
- Visualizing the result using graphs
- Selecting the best model and making the final predictions